

1 **§ 103 Rejections**

2 **Claims 1, 3-16 and 18-42** stand rejected under 35 U.S.C. § 103(a) over a
3 publication by Willie entitled “Presenting C#” (hereinafter “Willie”) in view of a
4 publication by Perry entitled “Teach Yourself Visual Basic 6 in 21 Days”
5 (hereinafter “Perry”).

6
7 **The Claims**

8 **Claim 1**, as amended [added language in bold italics], recites a software
9 architecture *embodied on one or more computer-readable storage media, the*
10 *software architecture* for a distributed computing system comprising:

- 11
- 12 • an application configured to handle requests submitted by remote
13 devices over a network; and
 - 14 • an application program interface *organized into multiple root*
15 *namespaces, the application program interface* to present
16 functions used by the application to access network and computing
17 resources of the distributed computing system, wherein calls to the
18 application program interface are handed to a common language
19 runtime layer *supporting* applications written in *one or more*
20 different languages *and translated* into an intermediate supported
21 language, the application program interface comprising various
22 types related to constructing user interfaces, *wherein the types*
23 *belong to a group assigned a group name associated with one of*
24 *the root namespaces, and wherein each of the types is referenced*
25 *by a hierarchical name comprising a top level identifier prefixed*
 to the group name assigned to the group.

21 In making out the rejection of this claim, the Office relies on the
22 teachings of Willie and Perry, and argues that it would have been obvious to
23 integrate the teachings of these references.
24
25

1 Applicant respectfully disagrees with the Office's argument and submits
2 that the Office has not established a *prima facie* case of obviousness.
3 Nevertheless, in the interest of advancing the prosecution of this claim,
4 Applicant has amended this claim to clarify, among other things, that the
5 application program interface recited in the claim is "organized into multiple
6 root namespaces" and that "the types belong to a group assigned a group name
7 associated with one of the root namespaces, and wherein each of the types is
8 referenced by a hierarchical name comprising a top level identifier prefixed to the
9 group name."

10 Neither Willie nor Perry discloses or suggests such subject matter.
11 Hence, for at least this reason, this claim is allowable.

12 **Claims 3-4 and 41** depend from claim 1 and are allowable as depending
13 from an allowable base claim. These claims are also allowable for their own
14 recited features which, in combination with those recited in claim 1, are neither
15 disclosed nor suggested in the references of record, either singly or in
16 combination with one another.

17 **Claim 5**, as amended [added language in bold italics], recites an
18 application program interface embodied on one or more computer readable
19 *storage* media, comprising: multiple types related to constructing user interfaces,
20 individual types *being associated with one or more groups and being referenced*
21 *by one or more hierarchical names, wherein each hierarchical name includes a*
22 *top level identifier prefixed to a group name assigned to one of the one or more*
23 *groups, the types* comprising classes which represent managed heap allocated
24 data that has reference assignment semantics, interfaces that define a contract that
25 other types can implement, delegates that are object oriented function pointers,

1 structures that represent static allocated data that has value assignment semantics
2 and enumerations which are value types that represent named constants, wherein
3 the application program interface is associated with a common language runtime
4 layer *supporting* applications written in *one or more of several* different
5 languages *and translated* into *an* intermediate language supported by the
6 common runtime layer.

7 In making out the rejection of this claim, the Office relies on the
8 teachings of Willie and Perry and argues it would have been obvious to
9 integrate the teachings of these references.

10 Applicant respectfully disagrees with the Office's argument and submits
11 that the Office has not established a *prima facie* case of obviousness.
12 Nevertheless, in the interest of advancing the prosecution of this claim,
13 Applicant has amended this claim to clarify, among other things, that individual
14 types are "associated with one or more groups" and are "referenced by one or
15 more hierarchical names, wherein each hierarchical name includes a top level
16 identifier prefixed to a group name assigned to one of the one or more groups".

17 Neither Willie nor Perry discloses or suggests such subject matter.
18 Hence, for at least this reason, this claim is allowable.

19 **Claims 6-15** depend from claim 5 and are allowable as depending from an
20 allowable base claim. These claims are also allowable for their own recited
21 features which, in combination with those recited in claim 5, are neither disclosed
22 nor suggested in the references of record, either singly or in combination with one
23 another.

24 **Claim 16**, as amended [added language in bold italics], recites a
25 distributed computer software architecture *embodied on one or more computer-*

1 *readable storage media, the distributed computer software architecture*

2 comprising:

- 3
- 4 • one or more applications configured to be executed on one or more
5 computing devices, the applications handling requests submitted from
6 remote computing devices;
- 7 • a networking platform to support the one or more applications;
- 8 • an application programming interface to interface the one or more
9 applications with the networking platform, the application
10 programming interface comprising various types related to constructing
11 user interfaces, *individual types being associated with one or more
12 groups and being referenced by one or more hierarchical names,
13 wherein each of the hierarchical names includes a top level identifier
14 prefixed to a group name assigned to one of the one or more groups;*
15 and
- 16 • a common language runtime layer *supporting* applications written in
17 *one or more* different languages *and translated* into an intermediate
18 language supported by the common runtime layer.

19 In making out the rejection of this claim, the Office relies on the
20 teachings of Willie and Perry and argues it would have been obvious to
21 integrate the teachings of these references.

22 Applicant respectfully disagrees with the Office's argument and submits
23 that the Office has not established a *prima facie* case of obviousness.
24 Nevertheless, in the interest of advancing the prosecution of this claim,
25 Applicant has amended this claim to clarify, among other things, that individual
types are "associated with one or more groups" and are "referenced by one or
more hierarchical names, wherein each of the hierarchical names includes a top
level identifier prefixed to a group name assigned to one of the one or more
groups".

Neither Willie nor Perry disclose or suggest such subject matter. Hence, for at least this reason, this claim is allowable.

Claims 18-27 and 42 depend from claim 16 and are allowable as depending from an allowable base claim. These claims are also allowable for their own recited features which, in combination with those recited in claim 16, are neither disclosed nor suggested in the references of record, either singly or in combination with one another.

Claim 28, as amended [added language in bold italics], recites a computer system including one or more microprocessors and one or more software programs, the one or more software programs utilizing an application program interface to request services from an operating system, the application program interface including separate commands to request services comprising services related to constructing user interfaces, wherein the application program interface groups API functions into multiple namespaces that define a collection of classes which represent managed heap allocated data that has reference assignment semantics, interfaces that define a contract that other types can implement, delegates that are object oriented function pointers, enumerations which are value types that represent named constants and structures that represent static allocated data that has value assignment semantics, the application program interface being associated with a common language runtime layer *supporting* applications written in *one or more* different languages *and translated* into an intermediate language supported by the common runtime layer.

In making out the rejection of this claim, the Office relies on the teachings of Willie and Perry and argues it would have been obvious to integrate the teachings of these references "to get a full understanding [of] how to construct

1 user interface of the client application.” However, the Office fails to address the
2 language “wherein the application program interface groups API functions into
3 multiple namespaces that define...value assignment semantics,” as also recited in
4 claim 28.

5 Applicant respectfully traverses this rejection and submits that the Office
6 has not established a *prima facie* case of obviousness.

7 First, Applicant submits that the references do not collectively disclose all
8 of the subject matter of claim 28. For example, Applicant fails to understand how
9 the “RequestWebPage”, an exemplary class for retrieving a web page, in Willie
10 can be equated with an “an application program interface to request services from
11 an operating system, the application program interface including separate
12 commands to request services comprising services related to constructing user
13 interfaces” as claimed.

14 Furthermore, it appears that the Office has forgotten the claim language
15 “wherein the application program interface groups API functions into multiple
16 namespaces that define...value assignment semantics”. Nevertheless, Applicant
17 has thoroughly reviewed the Willie and Perry references and submits that this
18 subject matter is neither disclosed nor suggested in the references of record, either
19 singly or in combination with one another.

20 Second, even if the cited references did disclose all the claimed subject
21 matter, which they do not, the Office’s stated motivation “to get a full
22 understanding [of] how to construct user interface of the client application,” is too
23 general and could not serve as the basis for making any modification to Willie.
24 Furthermore, this stated motivation is simply inapplicable to Willie, which is
25 directed to programming in a language different than that of Perry. As such, this

1 stated motivation simply fails to explain why one skilled in the art would have
2 been motivated to combine the teachings of the references. Therefore, Applicant
3 respectfully submits that that a person of ordinary skill in the art, at the time of
4 the invention, would not have reasonably looked to the teachings of Perry to
5 modify Willie.

6 In view of the above discussion, the Office has not established a *prima*
7 *facie* case of obviousness. Accordingly, for at least this reason, Applicant
8 traverses this rejection and submits that claim 28 is allowable.

9 **Claim 29**, as amended [added language in bold italics], recites a method,
10 comprising:

- 11
- 12 • managing network and computing resources for a distributed
computing system;
- 13 • exposing a set of functions that enable developers to access the
14 network and computing resources of the distributed computing system,
the set of functions comprising functions to facilitate construction of
15 user interfaces, wherein the functions are grouped into multiple
namespaces that define a collection of classes which represent
16 managed heap allocated data that has reference assignment semantics,
interfaces that define a contract that other types can implement,
17 delegates that are object oriented function pointers, enumerations
which are value types that represent named constants and structures
18 that represent static allocated data that has value assignment semantics;
and
- 19 • using a common language runtime layer *supporting* applications
20 written in *one or more* different languages *and translated* into an
intermediate language supported by the common runtime layer.
- 21

22 In making out the rejection of this claim, the Office simply indicates
23 "see rejection of claim 5 above." Applicant respectfully traverses this rejection
24 and submits that the Office has not established a *prima facie* case of obviousness.
25

1 First, Applicant submits that the references do not collectively disclose all
2 of the subject matter of this claim. For example, Applicant submits that claim 5
3 does not recite "wherein the functions are grouped into multiple namespaces ...
4 "as recited in claim 29. Accordingly, it appears that the Office has forgotten the
5 claim language "wherein the application program interface groups API functions
6 into multiple namespaces that define...value assignment semantics".
7 Nevertheless, Applicant has thoroughly reviewed the Willie and Perry references
8 and submits that this subject matter is neither disclosed nor suggested in the
9 references of record, either singly or in combination with one another.

10 Furthermore, Applicant is unable to find any mention of "classes which
11 represent managed heap allocated data that has reference assignment semantics"
12 on pages 48-49 of Willie or "interfaces that define a contract that other types can
13 implement" on pages 49-50 of Willie.

14 In addition, as noted above, even if the cited references did disclose all the
15 claimed subject matter, which they do not, the Office's stated motivation is too
16 general and is simply inapplicable to Willie. Therefore, Applicant respectfully
17 submits that a person of ordinary skill in the art, at the time of the invention,
18 would not have reasonably looked to the teachings of Perry to modify Willie.

19 Accordingly, in view of the above discussion, the Office has not
20 established a *prima facie* case of obviousness. Hence, for at least these
21 reasons, this claim is allowable.

22 **Claim 30** depends from claim 29 and is allowable as depending from an
23 allowable base claim. This claim is also allowable for its own recited features
24 which, in combination with those recited in claim 30, are neither disclosed nor
25

1 suggested in the references of record, either singly or in combination with one
2 another.

3 **Claim 31** recites a method, comprising creating a namespace with
4 functions that enable drawing and construction of user interfaces, the name space
5 defining classes which represent managed heap allocated data that has reference
6 assignment semantics, interfaces that define a contract that other types can
7 implement, delegates that are object oriented function pointers, structures that
8 represent static allocated data that has value assignment semantics, and
9 enumerations which are value types that represent named constants.

10 In making out the rejection of this claim, the Office relies on the teachings
11 of Willie and Perry and argues that it would have been obvious to integrate the
12 teachings of these references “to get a full understanding [of] how to construct
13 user interface of the client application.”

14 Applicant respectfully traverses this rejection and submits that the Office
15 has not established a *prima facie* case of obviousness.

16 First, Applicant submits that the references do not collectively disclose all
17 of the subject matter of this claim. For example, the Office’s reliance on pages
18 117-118 of Willie for disclosing “creating a namespace with functions that enable
19 drawing and construction of user interfaces, the name space defining classes
20 which represent managed heap allocated data that has reference assignment
21 semantics, interfaces that define a contract that other types can implement,
22 delegates ... structures ... and enumerations ...” is misplaced because these pages
23 merely describe creating namespaces in general, but do not describe creating a
24 namespace defining the subject matter of claim 31.
25

1 Furthermore, Applicant is unable to find any mention of “classes which
2 represent managed heap allocated data that has reference assignment semantics”
3 on pages 48-49 of Willie or “interfaces that define a contract that other types can
4 implement” on pages 49-50 of Willie.

5 Second, as noted above, even if the cited references did disclose all the
6 claimed subject matter, which they do not, the Office’s stated motivation is too
7 general and is simply inapplicable to Willie. Therefore, Applicant respectfully
8 submits that that a person of ordinary skill in the art, at the time of the invention,
9 would not have reasonably looked to the teachings of Perry to modify Willie.

10 Accordingly, in view of the above discussion, the Office has not
11 established a *prima facie* case of obviousness. Hence, for at least these
12 reasons, this claim is allowable.

13 **Claims 32-40** depend from claim 31 and are allowable as depending
14 from an allowable base claim. These claims are also allowable for their own
15 recited features which, in combination with those recited in claim 31, are
16 neither disclosed nor suggested in the references of record, either singly or in
17 combination with one another:

18 **Conclusion**

19 Applicant respectfully submits that the Office has failed to establish a
20 *prima facie* case of obviousness for the reasons set forth above. Applicant
21 respectfully requests a Notice of Allowability be issued forthwith.
22
23
24
25

Respectfully submitted,

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
Dated:

7/22/07

By:



Rich Bucher
Reg. No. 57,971
(509) 324-9256 ext. 216